In the Claims:

Please amend Claims 1-8, 15, 17, 19-22, 28-29, and 31-33, and add new Claim 34, all as shown below. Applicant respectfully reserves the right to prosecute any originally presented or canceled claims in a continuing or future application.

1. (Currently Amended) ~~An interactive tool~~ A system for supporting application deployment ~~manipulating a plurality of deployment descriptors~~, comprising:

a plurality of deployment descriptors that are adapted to describe deployment and configuration information of a plurality of applications deployed on a web server, wherein each [[one]] application of the plurality of applications is associated with at least one deployment descriptor of the plurality of deployment descriptors ~~that describes deployment and configuration information of the application on the web server~~; and

a builder component capable of

creating a master tree data structure based on [[the]] a present state of all deployment descriptor files, wherein the master tree data structure represents ~~a state of the logical hierarchy of~~ resources associated with the plurality of applications ~~at a given time~~, wherein the plurality of applications include a first application and the master tree data structure includes a sub-tree that corresponds to the first application;

creating a separate tree data structure ~~that represents deployment descriptor information~~ based on [[the]] a current state of source files in [[an]] a project directory associated with the first application's ~~project directory~~, wherein the separate tree data structure represents ~~a state of the logical hierarchy of~~ resources associated with the first application~~, wherein the application is one of the plurality of applications deployed on the web server~~;

comparing the sub-tree that corresponds to the first application in the master tree data structure with the separate tree data structure; and

refreshing the master tree data structure based on the separate tree data structure, if the master tree data structure is different from the separate tree data structure.

2. (Currently Amended) The ~~interactive tool~~ system of claim 1, further comprising:

a user interface capable of rendering an error message.

3. (Currently Amended) The ~~interactive tool~~ system of claim 2 wherein:

user selection of the error message can cause [[the]] a second user interface to render a user-editable representation of the at least one deployment descriptor component that is in error.

4. (Currently Amended) The ~~interactive tool~~ system of claim 1, further comprising:

a parser capable of generating a representation of the at least one deployment descriptor;

a generator capable of creating the at least one deployment descriptor; and

a validator capable of validating the at least one deployment descriptor.

5. (Currently Amended) The ~~interactive tool~~ system of claim 4 wherein:

the validator is capable of generating an error when it encounters a syntactic or semantic fault in the at least one deployment descriptor.

6. (Currently Amended) The ~~interactive tool~~ system of claim 1, wherein:

the builder component is further capable of automatically updating the at least one deployment descriptor to reflect one or more changes in at least one source code file.

7. (Currently Amended) The ~~interactive tool~~ system of claim 1 wherein:

the hierarchical representation can include information pertaining to an archive file.

8. (Currently Amended) The ~~interactive tool~~ system of claim 1 wherein:

the at least one deployment descriptor can be expressed as an Extensible Markup Language document.

9-14. (Canceled).

15. (Currently Amended) A method for supporting application deployment ~~providing an interactive tool for manipulating a plurality of deployment descriptors~~, comprising:

deploying a plurality of applications on a web server, wherein each [[one]] application of the plurality of applications is associated with at least one deployment descriptor of [[the]] a plurality of deployment descriptors that describes deployment and configuration information of the application on the web server;

creating a master tree data structure based on [[the]] a present state of all deployment descriptor files, wherein the master tree data structure represents ~~a state of the logical hierarchy of~~ resources associated with the plurality of applications ~~at a given time~~, wherein the plurality of applications include a first application and the master tree data structure includes a sub-tree that corresponds to the first application;

creating a separate tree data structure ~~that represents deployment descriptor information~~ based on [[the]] a current state of source files in [[an]] a project directory associated with the first application~~'s project directory~~, wherein the separate tree data structure represents ~~a state of the logical hierarchy of~~ resources associated with the first application~~, wherein the application is one of the plurality of applications deployed on the web server~~;

comparing the sub-tree that corresponds to the first application in the master tree data structure with the separate tree data structure; and

refreshing the master tree data structure based on the separate tree data structure, if the master tree data structure is different from the separate tree data structure.

16. (Original) The method of claim 15, further comprising:

providing a parser capable of generating a representation of the at least one deployment descriptor;

providing a generator capable of creating the at least one deployment descriptor; and

providing a validator capable of validating the at least one deployment descriptor.

17. (Currently Amended) The method of claim 16 ~~wherein~~ , further comprising:

~~the validator is capable of~~ generating, via the validator, an error when it encounters a syntactic or semantic fault in the at least one deployment descriptor.

18. (Original) The method of claim 15, further comprising:

providing a builder component capable of automatically updating the at least one deployment descriptor to reflect one or more changes in at least one source code file.

19. (Currently Amended) The method of claim 15 wherein:

including the hierarchical representation ~~can include~~ information pertaining to an archive file.

20. (Currently Amended) The method of claim 15 wherein:

expressing the at least one deployment descriptor ~~can be expressed~~ as an Extensible Markup Language document.

21. (Currently Amended) A machine readable medium having instructions stored thereon that when executed by a processor cause a system to:

deploy a plurality of applications on a web server, wherein each [[one]] application of the plurality of applications is associated with at least one deployment descriptor of a plurality of deployment descriptors that describes deployment and configuration information of the application on the web server;

create a master tree data structure based on [[the]] a present state of all deployment descriptor files, wherein the master tree data structure represents ~~a state of the logical hierarchy of~~ resources associated with the plurality of applications ~~at a given time~~, wherein the plurality of applications include a first application and the master tree data structure includes a sub-tree that corresponds to the first application;

create a separate tree data structure ~~that represents deployment descriptor information~~

based on [[the]] a current state of source files in [[an]] a project directory associated with the first application~~'s project directory~~, wherein the separate tree data structure represents ~~a state of the logical hierarchy of~~ resources associated with the first application~~, wherein the application is one of the plurality of applications deployed on the web server~~;

compare the sub-tree that corresponds to the first application in the master tree data structure with the separate tree data structure; and

refresh the master tree data structure based on the separate tree data structure, if the master tree data structure is different from the separate tree data structure.

22. (Currently Amended) The machine readable medium of claim 21, further comprising instructions that when executed cause the system to:

provide a user interface capable of rendering an error message, wherein user selection of the error message in [[the]] a third user interface can cause [[the]] a second user interface to render a user-editable representation of the at least one deployment descriptor component that is in error.

23. (Original) The machine readable medium of claim 21, further comprising instructions that when executed cause the system to:

provide a parser capable of generating a representation of the at least one deployment descriptor;

provide a generator capable of creating the at least one deployment descriptor; and

provide a validator capable of validating the at least one deployment descriptor.

24. (Original) The machine readable medium of claim 23 wherein:

the validator is capable of generating an error when it encounters a syntactic or semantic fault in the at least one deployment descriptor.

25. (Original) The machine readable medium of claim 21, further comprising instructions that when executed cause the system to:

provide a builder component capable of automatically updating the at least one deployment descriptor to reflect one or more changes in at least one source code file.

26. (Previously Presented) The machine readable medium of claim 21 wherein:
the hierarchical representation can include information pertaining to an archive file.

27. (Original) The machine readable medium of claim 21 wherein:
the at least one deployment descriptor can be expressed as an Extensible Markup Language document.

28. (Currently Amended) The ~~interactive tool~~ system of claim 1, wherein:
the interactive tool is capable of automatically repairing a first deployment descriptor of the at least one deployment descriptors if the first deployment descriptor is defective.

29. (Currently Amended) The ~~interactive tool~~ system of claim 1, wherein:
the builder component is further capable of creating a tree data structure that embodies hierarchical relationships of nested XML statements.

30. (Canceled).

31. (Currently Amended) The ~~interactive tool~~ system of claim 1, wherein:
the builder component is further capable of
allowing a module to be shared by both the first application and a second application;
disassociating the module from the first application in the master tree data structure, when the module is removed from the first application;
keeping [[a]] the module in the master tree data structure to allow ~~applications other than a current application~~ the second application to use the

module~~, even after the module is removed from the current application~~.

32. (Currently Amended) The ~~interactive tool~~ system of claim 1, further comprising:

a first user interface capable of rendering a hierarchical representation of the plurality of deployment descriptors, wherein a component of the representation can be selected by a user; and

a second user interface capable of rendering a user-editable representation of the selected component.

33. (Currently Amended) The ~~interactive tool~~ system of claim 1, wherein:

the builder component is further capable of generating a new deployment descriptor for the application from the refreshed master tree data structure.

34. (New) The system of claim 1, further comprising:

a pane that displays a single field for a value, wherein the single field maps to multiple values in the at least one deployment descriptor.